#1)QUESTION

```
n1=int(input("Enter the number of elements in the first list"))
l1=[]
for i in range(n1):
    name=input()
    l1.append(name)
n2=int(input("Enter number of elements in the second list"))
l2=[]
for i in range(n2):
    grade=int(input())
    l2.append(grade)
for i in range(min(len(l1),len(l2))):
    print(l1[i],l2[i])
```

#2) QUESTION

```
import array as arr
array_list=arr.array('i')
n=int(input("Enter the number of elements in the array"))
for i in range(n):
    elements=int(input())
    array_list.append(elements)
print(array_list)#34,45,23,60
minimum=array_list[1]#45
for i in array_list:
    if i<minimum:
        minimum=i#34#23
print("Minimum element=",minimum)
```

```
#3)QUESTION
import array as arr
array_list=arr.array('i')
array_list2=arr.array('i')
n1=int(input("Enter number of elemnts in array1"))
for i in range(n1):
    elements=int(input())
    array_list.append(elements)
n2=int(input("Enter number of elememts in array2"))
for i in range(n2):
    components=int(input())
    array_list2.append(components)
l1=array_list.tolist()
l2=array_list2.tolist()
l1.extend(l2)
l1.sort(reverse=True)
new_array=arr.array('i',l1)
for i in new_array:
    print(i,end=" ")



#4
Name=str(input("Name:"))
Age=int(input("Age:"))
CGPA=float(input("CGPA:"))
Grade=input("Grade:")
print("Name:", Name)
print("Age:", Age)
print(f"CGPA: {math.trunc{CGPA:.2f}}")
print("Grade:", Grade)
```

```
#5
radius=int(input("enter your radius:"))
Area=3.14*radius*radius
print(Area)

#6)question PALINDROME:

n=int(input("Enter a number to check palindrome"))
num=n
rev=0
while n!=0:
    temp=n%10
    rev=(rev*10)+temp
    n=n//10
if num==rev:
    print(f"{num} is a palindrome number")
else:
    print(f"{num} is not  a palindrome number")


#7)QUESTION SERIES:


import math
x=int(input("Enter the number for series"))
n=int(input("Enter the power"))
sum_series=1
while n>0:
    term=math.pow(x,n)
```

```
    sum_series+=term/math.factorial(n)
    n=n-1
print(sum_series)
```

#8)QUESTION PRODUCT OF THE DIGITS:

```
n=int(input("Enter the number to find the product of the digits "))
n=str(n)
product=1
for i in n:
    i=int(i)
    product=product*i
print("The product of the digits is ",product)
```

#9)QUESTION FACTORIAL OF THE NUMBER:

```
n=int(input("Enter the number for printing the factorial"))
factorial=1
while n>1:
    factorial*=n*(n-1)
    n=n-2
print("The factorial is ",factorial)
```

#10) QUESTION PRIME NUMBER OR NOT

```
n=int(input("Enter the number for finding prime number"))
if n<2:
    flag=False
if n==2:
      flag=True
if n>2:
```

```
    for i in range(2,n):
        if n%i==0:
            flag=False
            break
        else:
            flag=True
if flag:
    print("The number is prime")
else:
        print("The number is not prime")
```

#11)write the output of the following

```
for i in range (5,9):
    print(i)
```

#Output

#5

#6

#7

#8

#12 Write the output of the following:

```
for i in "Python":
    print(i,end=" ")
```

#Output:

```
# P y t h o n


#13)Write the output of the following:
for i in "Python":
    print(i,'?$')


#Output:
#P?$
#y?$
#t?$
#h?$
#o?$
#n?$



#14)WRITE THE OUTPUT OF THE FOLLOWING:
X=8
while x<=20:
    print(x**2)
    x+=3
#output:
#64
#121
#196
#289
#400



#15)QUESTION PRINT THE FOLLOWING PATTERN:
for i in range(1,5):
```

```
    for j in range(1,i+1):
        print(j,end=" ")
    print()
```

#16)Print the numbers that are beteween 100 and 500that are divisible by 11 but not divisible by 2

```
n=100
while n<500:
    if n%11==0 and n%2!=0:
        print(n,end=" ")
    n=n+1
```

#17

```
L=[12,34,56,67]
print(L)
L.append(90)
print(L)
L.remove(56)
print(L)
```

#18

```
account_name=input("enter the account name:")
account_number=int(input("enter the account number:"))
balance=float(input("enter the balance amount:"))
print("Account Name:", account_name)
print("Account Number:", account_number)
print(f"Balance: {balance:.2f}")
```

#19

```
a=int(input("enter your age:"))
print("eligible"if a>=24 else "not eligible")

#20
l1=int(input())
l2=int(input())
l3=int(input())
l4=int(input())
l5=int(input())
L=list([l1,l2,l3,l4,l5])
print(L)
L[2]=34
print(L)

#21
txt="harini is good looking,and she is caring"
print("islower:",txt.islower())#check whether it is in lower
print("isprintable:",txt.isprintable())#list,tuple cannot be printed or any other special characters
print("isnumeric:",txt.isnumeric())#check whether it is number
print("isspace:",txt.isspace())#check whether spaces before first string
print("istitle:",txt.istitle())#check whether the first char is in caps
print("isupper:",txt.isupper())#check whether it is in upper case
print("join:",txt.join("and"))#join the given like aharini is good looking,and she is caringnharini is good looking,and she is caringd
print("ljust:",txt.ljust(50,"$"))#harini is good looking,and she is caring$$$$$$$$$$
print("lower:",txt.lower())#returns the lower case
print("lstrip:",txt.lstrip())#trims towards the left
mytrans=str.maketrans("good","GOOD")#EVERY O AND D AND G WILL CHANGE
print("translate:",txt.translate(mytrans))#translate the given
print("partition:",txt.partition("good"))#it will separate after good
print("replace:",txt.replace("g","G"))#replace the string speaking only
```

```
print("rfind:",txt.rfind("is good"))#find the last occurence of the given word

print("rjust:",txt.rjust(50,"#"))#fill with # for 50

print("rpartition:",txt.partition("speaking"))#patition after speaking

print("rsplit:",txt.rsplit())#split

print("rstrip:",txt.rstrip())#STRIP TRIM TO RIGHT

print("split:",txt.split())#SPLIT THE STRING

print("splitlines:",txt.splitlines())#SPLIT THE LINES

print("startswith:",txt.startswith("s")#CHECK WHETHER STARTS WITH S)

print("title:",txt.title()) #CHECK WHTEHER TITLE


print("upper:",txt.upper())#CHANGES INTO UPPER

print("zfill:",txt.zfill(50))#FILL 0S IN FRONT

print("identifier:",txt.isidentifier())#check whether alpha numeric is present or underscore is present

print(txt.rindex("is good"))#it will return the last occourence index of the given string ex is means last index is 31

print(txt.find("is"))#it will give the first occurence of the string


#22


l=[10,45,1,67,56]

print(l)

l.sort()

print("list in ascending order:",l)

l.sort(reverse=True)

print("list in descending order:",l)


#23

s1={10,20,45,67}

s2={45,67,30,29}

print("the set 1:",s1)

print("the set 2:",s2)
```

```
s3=s1.intersection(s2)
print("intersection:",s3)
s4=s1.symmetric_difference(s2)
print("symmetric difference:",s4)
s1.remove(45)
print("after removing:",s1)

#24
t=(10,20,30)
print("the tuple is:",t)
l=list(t)
l.append(90)
t=tuple(l)
print("after adding:",t)
l=list(t)
l.remove(20)
t=tuple(l)
print("after removing:",t)

#25
a="     I am studying python"
print("the string is:",a)
print(a.rindex("n"))
print("replace:",a.replace("a","A"))
print("lstrip:",a.lstrip())



#26)
import array as arr
array=[]
```

```
negative=[]
positive=[]
n=int(input("Enter the number of elemnts in the array"))
for i in range(n):
    temperatures=int(input())
    array.append(temperatures)
for i in array:
    if i<0:
        negative.append(i)
    else:
        positive.append(i)
for i in negative:
    print(i,end=" ")
for i in positive:
    print(i,end=" ")
print()
#27)

for i in range(65,70):
    for j in range(65,i+1):
        print(chr(j),end=" ")
    print()

#28)

for x in range(10,20):
    if x%2==0:
        continue
        print(x)
```

OUTPUT :BLANK LINE WILL BE PRINTED SINCE PRINT(X) INSIDE THE LOOP IF IT IS OUTSIDE THE LOOP

13

15

17

19 WILL BE PRINTED

```
#29)
for x in range(3,9,2):
    print(x*10)
#30)
import array as arr
n=int(input())
array=[]
list_elements=[]
new_array=[]
for i in range(n):
    user_id=int(input())
    array.append(user_id)
for i in array:
    if i not in list_elements:
        list_elements.append(i)
    else:
        print(i)

#31)another method
n=int(input())
a=[]
for i in range(n):
    a.append(int(input()))
j=0
for i in range(0,n):
    if(a[i]<0):
```

```
    t=a[i]
    a[j]=t
    j=j+1
print(a)



#32)QUESTION
elements=list(map(int,input().split()))
largest=min(elements)#1
second_largest=min(elements)#1
for i in elements:#5#4#3
   if i>largest:
      second_largest=largest#5
      largest=i#4
   elif i>second_largest and i<largest:
         second_largest=i#4
print(second_largest)#4

#33)QUESTION
N=int(input())
count=0
array=[]
n=len(array)
for i in range(N):
   elements=int(input())
   array.append(elements)
majority_element=int(input())
occurence=array.count(majority_element)
if occurence>n/2:
   print(majoity_element,"is a majority element")
else:
```

```
    print(majority_element,"is not a majority element")

#34)QUESTION
phone_number=input()
replaced=input()
new=input()
new_num=phone_number.replace(replaced,new)
print(new_num)

#35)QUESTION
log_in={"harini":12345,"poorani":34567,"rickshitha":67895,"anika":9086745}
user_name=input()
user_password=int(input())
if user_name in log_in:
    if log_in[user_name]==user_password:
        print("Successfully logged in")
    else:
        print("Incorrect password")
else:
    print("Incorrect user")


#36)

set_items=[]
set_items2=[]
n1=int(input("Enter the number of element in the set:"))
for i in range(n1):
    elements=input()
    set_items.append(elements)
set_items=set(set_items)
```

```
print(set_items)
n2=int(input("Enter the number of elements in set 2"))
for i in range(n2):
    elements=input()
    set_items2.append(elements)
set_items2=set(set_items2)
print(set_items2)
if n1>n2:
    for i in set_items2:
        if i  in set_items:
            subset=True
        else:
            subset=False
            break

    print("Is set 2 is subset?")
if n2>n1:
    for i in set_items:
        if i in set_items2:
            subset=True
        else:
            subset=False
            break
    print("Is set 1 is subset?")
if subset:
    print("yes subset")
else:
    print("No")

#37)
tuple_items=('1','2','4','5','6','7','9')
```

```
list_items=list((tuple_items))
sliced=list_items[2:4]
for i in sliced:
    print(i)

#38)
tuple_items=('1','2','harini','bsc')
a,b,c,d=tuple_items
print(a)
print(b)
print(c)
print(d)

#39)
list_items=[(2, 5), (9, ), (8, 7, 6), (4, ), (12, 4, 16, 7)]
new_list=[]
for i in list_items:
    if len(i)<3 or len(i)>3:
        new_list.append(i)
print(new_list)


#40)
string=("my","pyhton","programmming")
new_string=" "
for i in string:
    print(i+new_string,end=" ")
print()

#41)
item=(("name","harini"),("age",18),("standard","bsc"))
```

```
print(dict(item))


#42)
for i in range(5,0,-1):
    for j in range(5-i):
        print(' ',end=" ")

    for k in range(i):
        print("*",end=" ")
    print()
for i in range(5-1):
    print(' ',end=" ")
print("*")
for i in range(2,5+1):
    for j in range(5-i):
        print(' ',end=" ")

    for k in range(i):
        print("*",end=" ")
    print()

#43)prime number or not
n=int(input("Enter a number to check whether prime number or not"))
count=0
if n<=1:
    print("The given number is neither prime nor composite number")
else:
    for i in range(1,n+1):
        if n%i==0:
            count+=1
    if count==2:
```

```
        print("The given number is prime number ")
    else:
        print("the given number is not a prime number")

#44) Factorial of a given number:
num=int(input("Enter the number to find the factorial"))
fact=1
while num>1:
    fact*=num*(num-1)
    num=num-2
print("The factorial of the given number is ",fact)

#45)fibbonacci series:
n=int(input("Enter a number to find fibonnaci series"))
n1=0
n2=1
count=0
if n==0:
    print("enter a valid number")
elif n==1:
    print(n1)
else:
    while count<n:
        print(n1,end=" ")
        n1,n2=n2,n1+n2
        count+=1

#45)divisibility by 11
n=int(input("Check whether a number is divisible by 11 or not"))
if n%11==0:
    print("The given number is divisible by 11")
```

```
else:
    print("the given number is not divisible by 11")



#47)factors of a number
n=int(input("enter the number to find its factors"))
for i in range(1,n+1):
    if n%i==0:
        print(i,end=" ")



#48)minimum of two numbers
a=int(input())
b=int(input())
min=a if a<b else b
print(min)



#49)compound interest
import math
p=float(input())
r=float(input())
r_interest=r/100
t=float(input())
amount=p*(1+r_interest)**t
compound_interest=amount-p
print(math.trunc(compound_interest))



#50)swapping of two numbers
```

```
a=int(input())
b=int(input())
l=list((a,b))
l.reverse()
a=l[0]
b=l[1]
print(f"{a} {b}")

#51)creating list

l=[]
l.extend([1,2,"apple",3,4])
search=input()
if search in l:
    print("yes")
else:
    print("no")

#52) checking wheteher the search is not present or present
s="hello,welcome to python programming"
search=str(input())
if search not in s:
    print("true")
else:
    print("false")


#53)book programme

n=int(input("enter the number of books"))
l=[]
```

```
for i in range(n):
    books=input()
    l.append(books)
print("1)display 2)add 3)remove")
while True:
    option=int(input("enter the option"))
    match option:
        case 1:
            def display():
                for i in l:
                    print(i)
            display()
        case 2:
            def add_books():
                add=input("Enter the new book to be added")
                l.append(add)
                print("list of books after adding:",l)
            add_books()
        case 3:
            def remove_books():
                remove=input("enter the book to be removed")
                l.remove(remove)
                print("list after removal of book",l)


            remove_books()


        case _:
            break
```

#54)

```
d={"product 1":120,"product2":100,"product 3":50}
print("1)display 2)add 3)remove")
while True:
   option=int(input("enter the option"))
   match option:
      case 1:
         def display():
            for i in d.items():
               print(i)
         display()
      case 2:
         def add_new():
            product=input("enter the new product name")
            quantity=int(input())
            d[product]=quantity
            print("after adding",d)
         add_new()
      case 3:
         def remove():
            remove=input("enter the discontinued product")
            if remove in d:
               del d[remove]
               print("after removing",d)
         remove()
      case _:
         quit()
```

#55)checking whteher odd or even:

```
a=int(input())
if a%2==0:
    print("even")
else:
    print("odd")
```

#56)check whether positive,negative or zero:

```
a=int(input())
if a==0:
    print("Zero")
elif a>0:
    print("Positive")
else:
    print("Negative")
```

#57)checking a student's eligiblity to join a course:

```
p=int(input())
c=int(input())
m=int(input())
total=p+c+m
print(total)
percent=((p+c+m)/300)*100
print(f"{percent:.2f}")
if percent>90:
    print("Eligible")
else:
    print("Not Eligible")
```

#58) check whether the given string is in upper or lower case or in both:

```
s=str(input())
if s.isupper():
   print("upper")
elif s.islower():
      print("lower")
else:
   print("combination of both")
```

#59)checking of divisiblit by 3 or 5

```
n=int(input())
if n%3==0:
   if n%5==0:
      print("Divisible by both")
   else:
      print("Divisible by 3")
elif n%5==0:
   print("Divisible by 5")
else:
   print("Not divisible by both :")
```

#60)paying bill

```
import math
```

```
cost_1=int(input())
cost_2=int(input())
cost_3=int(input())
total=cost_1+cost_2+cost_3
if total>=400:
    tax=math.trunc((6.75/100)*total)
    final_amount=tax+total
    tips=math.trunc((10/100)*final_amount)
    total_paying=math .trunc(final_amount+tips)
    print(total)
    print(tax)
    print(tips)
    print(total_paying)
else:
    print(total)
```

#61)calculate bmi

```
weight=float(input("enter weight in kg:"))
height=float(input("enter height in m:"))
BMI=(weight/height**2)
print(f"{BMI:.1f}")
if BMI<16:
    print("Severe Thinness")
elif BMI>=16 and BMI<17:
    print("Moderate Thinness")
elif BMI>=17 and BMI<18.5:
    print("Mild Thinness")
```

```
elif BMI>=18.5 and BMI<25:
   print("Normal")
elif BMI>=25 and BMI<30:
   print("Overweight")
elif BMI>=30 and BMI<35:
   print("Obese Class I")
elif BMI>=35 and BMI<40:
   print("Obese Class II")
elif BMI>=40:
   print("Obese Class III")
else:
   print("incorrect info")
```

#62)QUESTION ACCEPTING DIFFERENT VALUE AS PARAMETER AND CONVERTING INTO A LIST AND RETURNING A LIST ELEMENTS:

```
def new_list():
   n=int(input("ENTER THE NUMBER OF ELEMENTS IN THE LIST"))
   l=[]
   for i in range(n):
      elements=int(input())
      l.append(elements)
   return l
l=new_list()
print("The new list is" ,l)
```

#63) PYTHON PROGARM TO FIND THE FACTORIAL OF THE NUMBER:

```
def fact(n):
   fact=1
   while n>1:
      fact*=n*(n-1)
```

```
        n=n-2
    return fact
n=int(input("ENTER THE NUMBER TO FIND THE FACTORIAL "))
ans=fact(n)

print("the factorial of the given number is ",ans)



#64)ACCEPTING TWO NUMBERS AND RETURNING THEIR PRODUCT:
def product(n1,n2):
    ans=a*b
    return ans
a=int(input())
b=int(input())
a=product(a,b)
print("the product of the given two numbers is",a)



#65)
def display(a,b):
    print(last_name,end=" ")
    print(first_name,end=" ")

first_name=input()
last_name=input()
display(first_name,last_name)

#66)FINDING THE SQUARE OF THE NUMBERS ENTERED BY THE USER
def square():
    a=int(input("enter the number to find its square"))
    squared=a*a
```

```
    return squared
ans=square()
print("the square of the given number is",ans)




#67)

def details(Name,Age,Department):
    print("Age:",Age)
    print("Name:",Name)
    print("Department:",Department)

details(Name="harini",Age=18,Department="computer science with artificial intelligence")

#68)
def myFavfood(hDict):
    print(hDict)
hfoodtype=input("enter the food type")
hitemname=input("enter the item name")
hitemcost=input("enter the item cost")
Food_items={
    'Foodtype':hfoodtype,
    'Itemname':hitemname,
    'Foodcost':hitemcost
    }
myFavfood(Food_items)

#70)
arr=[]
total=0#1 2 3
```

```
    #4 5 6
n=int(input("Enter the number of rows and coloumns:"))
for i in range(n):
   product=list(map(int,input().split()))
   arr.append(product)

for i in range(n):
   for j in range(n):
      print(arr[i][j],end=" ")
   print()
for i in range(n):
   for j in range(n):#0+1=1+2=3+3=6+4=10+5=15+6=21
      total+=arr[i][j]

print(total)

#71)
arr=[]
n=int(input("the number of elements in the array"))
element_found=int(input("enter the element to be found"))
element=False
for i in range(n):
   score=list(map(int,input().split()))
   arr.append(score)

for i in range(n):
   for j in range(n):
      print(arr[i][j],end=" ")
   print()
for i in range(n):
   for j in range(n):
```

```
        if element_found==arr[i][j]:
            element=True
            print("found")
            print(f"the element in ({i},{j})")

if element==False:
    print("not found")




#72)
arr=[]

n=int(input("Enter the number of rows in the elements"))
for i in range(n):
    product=list(map(int,input().split()))
    arr.append(product)
print(arr)
for i in range(n):

    for j in range(n):
        print(arr[j][i],end=" ")
    print()

#73)
row=3
coloumn=4
arr=[[0, 1, 0, 1],
     [1, 0, 1, 0],
     [0, 1, 1, 1]]
for i in range(row):
```

```
    for j in range(coloumn):
        if arr[i][j]==0:
            arr[i][j]=0
        else:
            arr[i][j]="x"
for i in range(row):
    for j in range(coloumn):
        print(arr[i][j],end=" ")
    print()


#74)
n=int(input("Enter the number of elements"))
arr=[]
for i in range(n):
    elements=list(map(int,input().split()))
    arr.append(elements)
for i in range(n):
    for j in range(n):
        if arr[i][j]==0:
            arr[i][j]=0
        else:
            arr[i][j]="x"
for i in range(n):
    for j in range(n):
        print(arr[i][j],end=" ")
    print()



#75
import random
```

```
while True:
    user_input=input("enter any word")
    match user_input:
        case "pick":
            answer=random.randrange(1,6)
            print(answer)
        case "exit":
            print("exit")
            quit()
        case _:
            print("key word not recognised")
            break
#76
import random
restaurant=["Sangeetha","thalapakkati","jalpaan","KFC","Dominoes","pizza hut","saravana bhavan"]
while True:
    user_input=input("enter any word")
    match user_input:
        case "pick":
            answer=random.choice(restaurant)
            print(answer)
        case "bye":
            quit()
        case _:
            print("invalid user input")
            break
#77
import re
import random
import string
```

```
user_input=input("enter any name")
def check():
    valid=False
    pattern=r'^[a-zA-Z0-9_]+$'
    if user_input[0].isalpha() and len(user_input)>5 and len(user_input)<15 and
re.match(pattern,user_input):
        valid=True
    if valid:
            print("valid user name")
    else:
        validate()
def validate():
    letters=string.ascii_letters
    numbers=string.digits

    final=letters+numbers+"_"
    user_name=random.choice(letters)
    for i in range(4,14):
        user_name+=''.join(random.choice(final))
    print(user_name)

check()



#78)
arr1=[]
arr2=[]

n1=int(input("Enter the number of rows and coloumns in the matrix1"))
n2=int(input("Enter the number of rows and coloumns  in the matrix2"))
add=[[0]*n1 for i in range(n1)]
```

```
for i in range(n1):
    e=list(map(int,input().split()))
    arr1.append(e)
for i in range(n2):
    e=list(map(int,input().split()))
    arr2.append(e)
for i in range(n1):
    for j in range(n2):

        add[i][j]+=arr1[i][j]+arr2[i][j]
for i in range(n1):
    for j in range(n1):
        print(add[i][j],end=" ")
    print()

#79)
arr=[]

n1=int(input("enter the number of rows and coloumns in the matrix"))
tra=[[0]*n1 for i in range(n1)]

for i in range(n1):
    e=list(map(int,input().split()))
    arr.append(e)
for i in range(n1):
    for j in range(n1):
        tra[i][j]=arr[j][i]

rotated=[[0]*n1 for i in range(n1)]#0 1 1 1

for i in range(n1):#0
```

```
    for j in range(n1):#0
        rotated[j][n1-i-1]=tra[i][j]#01=00 11=01 00=10 10=11
for i in range(n1):
    for j in  range(n1):
        print(rotated[i][j],end=" ")
    print()



#80)
arr=[]
n1=int(input())
for i in range(n1):
    e=list(map(int,input().split()))
    arr.append(e)
maxi=arr[0][0]
for i in range(n1):
    for j in range(n1):
        if arr[i][j]>maxi:
            maxi=arr[i][j]
print(maxi)
#81)
arr=[]
n1=int(input())
s=False
for i in range(n1):
    e=list(map(int,input().split()))
    arr.append(e)
for i in range(n1):
    for j in range(n1):
        if arr[i][j]==arr[j][i]:
            s=True
```

```
        else:
            s=False
            break

if s:
    print("symmentric matrix")
if not s:
    print("not a symmentric matrix")

#82


class Employee:
    def __init__(self,emp_id,emp_name,emp_salary,emp_dept):
        self.emp_id=emp_id
        self.emp_name=emp_name
        self.emp_salary=emp_salary
        self.emp_dept=emp_dept
    def assign_department(self,new_dept):
        self.emp_dept=new_dept
    def print_employee_details(self):
        print(f"The id is :{self.emp_id}")
        print(f"the name is :{self.emp_name}")
        print(f"the salary is :{self.emp_salary}")
        print(f"the department is: {self.emp_dept}")
    def calculate_salary(self,hoursworked):
        if hoursworked>50:
            over_time=hoursworked-50
            overtime_amount=over_time*(self.emp_salary/50)
            total_salary=self.emp_salary+overtime_amount
        else:
            total_salary=self.emp_salary
```

```
        return total_salary
employees=[
    Employee(123,"Harini",12000,"Developer"),
    Employee(124,"priya",12500,"Tester"),
    Employee(125,"Rajesh",50000,"Analyst"),
    Employee(126,"Raja",55000,"Manager")
    ]


employees[0].assign_department("HR")
employees[0].print_employee_details()
salary_with_overtime=employees[0].calculate_salary(55)
print(f"Total salary:{salary_with_overtime}")



#83)question


cosmetic_brands={"Lakme":["foundation","lipstic","kajal","nailpolish","compact powder"],
                 "mama_earth":["beauty cream","hair spray","facial pack","skin care pac","face wash"],
                 "maybelline":["mostraiser","pot_kajal","pac of lipstic","maskra" ,"eyelash"],
                  "mac":["concealer","eybrom trimmer","vaccinizer,nail" "polish remover","skin concealer"],
                  "dazzler":["primer","sunscreen","lip balm","manicure pack","panicure pack"]}
user_brand=input("which brand you have to choose?")
if user_brand in cosmetic_brands:
    print(f"Your brand is {user_brand}")
    for i in cosmetic_brands[user_brand]:
        print(i)
else:
    print("Sorry your brand is not available")
```

#84)question

```
a=int(input())
b=int(input())
print("options are:1)Add 2)Sub 3)Mul 4)Div 5)Exp")
option=input()
match option:
    case"1":
        print("Add")
        sum=a+b
        print(sum)
    case"2":
        print("Sub")
        dif=a-b
        print(dif)
    case"3":
        print("Mul")
        pro=a*b
        print(pro)
    case"4":
        print("Div")
        quo=a//b
        print(quo)
    case"5":
        print("Exp")
        pow=a**b
        print(pow)
    case _:
```

```
    print("You have selected an invalid option")
```

#85)question

```
cosmetic_brands={"Lakme":["foundation","lipstic","kajal","nailpolish","compact powder"],
            "mama_earth":["beauty cream","hair spray","facial pack","skin care pac","face wash"],
            "maybelline":["mostraiser","pot_kajal","pac of lipstic","maskra" ,"eyelash"],
             "mac":["concealer","eybrom trimmer","vaccinizer,nail" "polish remover","skin concealer"],
             "dazzler":["primer","sunscreen","lip balm","manicure pack","panicure pack"]}
user_brand=input("which brand you have to choose?")
if user_brand in cosmetic_brands:
   print(f"Your brand is {user_brand}")
   for i in cosmetic_brands[user_brand]:
      print(i)
else:
   print("Sorry your brand is not available")
```

#86)question

```
a=int(input())
b=int(input())
print("options are:1)Add 2)Sub 3)Mul 4)Div 5)Exp")
option=input()
match option:
   case"1":
      print("Add")
```

```
        sum=a+b
        print(sum)
    case"2":
        print("Sub")
        dif=a-b
        print(dif)
    case"3":
        print("Mul")
        pro=a*b
        print(pro)
    case"4":
        print("Div")
        quo=a//b
        print(quo)
    case"5":
        print("Exp")
        pow=a**b
        print(pow)
    case _:
        print("You have selected an invalid option")



#87
print("Option:1)Deposit 2)Withdraw amount 3)View Balance 4)Exit")
balance=15000
while True:
    option=input("Enter your option:")
    match option:
```

```
    case"1":
        print("Deposit")
        deposit=float(input("enter your deposit amount:"))
        print(f"your deposited amount is Rs.{deposit}")
        balance=deposit+balance
    case"2":
        print("withdraw amount:")
        withdrawel=float(input("enter how much you want to withdraw?"))
        if balance>=withdrawel:
            print(f"you have withdrawn:Rs.{withdrawel}")
            balance=balance-withdrawel
        else:
            print(f"sorry you have only{balance}")



    case"3":
        print("View balance:")
        print(f"You have available balance{balance}")
    case"4":
        print("Exit")
        break
    case _:
        print("Invalid option:")
```

#88

```
menu_list={"Breakfast":["idly.....Rs.30","dosa.....Rs.60","poori set.....Rs.80","pongal set.....Rs.100"],
        "lunch":["Sambar rice.....Rs.120", "bisbellabath.....Rs.150","fried rice.....Rs.220","veg biriyani....Rs.300"],
        "dinner":["butter nan with garvy.....Rs.300","paratha with gravy.....Rs.320","onion kulcha.....Rs 200","Noodles.....Rs.200"],
        "desserts":["vennila ice cream.....Rs 150","chocolate ice cream.....Rs.150","butterscotch ice cream.....Rs.110"],
        "special offers":["breakfast any 1 with desserts sets.....Rs .220","dinners any 1 with desserts set.....Rs.240"]}
user_input=input("Choose any one menu:")
if user_input in menu_list:
    print(f"You have selected {user_input}")
    for i in menu_list[user_input]:
        print(i)
else:
    print("Sorry the menu is not available today!")
```

#89

```
for x in range(0,29,4):
    print(x,end=" ")
```

#90

```
rows=int(input())
coloumns=int(input())
for i in range (rows):
    for j in range (coloumns):
        print("&",end=" ")
```

```
    print()



#91


marks=[90,40,-5,80,-1]
for i in marks:
    if i>0:
        print(f"Score:{i}")
    if i <-1:
        print(f"Invalid score {i} encounterd")
        continue


    if i==-1:
        print("Encountered missing data .Stop processing")
        break




#92
class Employee:
    def __init__(self,name,salary):
        self.name=name
        self.salary=salary
    def display(self):
        print(f"Name:{self.name}\nSalary:{self.salary}")
class Manager(Employee):
    def __init__(self,name,salary,department):
        super(). __init__(name,salary)
        self.department=department
    def printDetails(self):
```

```
        self.display()
        print(f"Department:{self.department}")
s=Manager('Harini',20000,'IT')
s.printDetails()
print()

#93
class Library:
    def __init__(self,title,author,publication_year):
        self.title=title
        self.author=author
        self.publication_year=publication_year
    def display(self):
        print(f"Title:{self.title}\nAuthor:{self.author}\nPublication year:{self.publication_year}")
class Book(Library):
    def __init__(self,title,author,publication_year,genre):
        super(). __init__(title,author,publication_year)
        self.genre=genre
    def printDetails(self):
        self.display()
        print(f"Genre:{self.genre}")
s=Book('Ponniyin selvan','Kalki',1967,'History')
s.printDetails()
print()
#94
class Bank:
    def __init__(self):
        self.balance=0
    def deposit(self,deposit):

        self.balance+=deposit
```

```
        print("successfully deposited")
    def withdraw(self,withdraw):

        if self.balance>withdraw:
            self.balance-=withdraw
            print("you have successfully withdrawn")
        else:
            print("no sufficient balance")
    def bal(self):
        print(f"Balance:{self.balance}")
s=Bank()
s.deposit(10000)
s.withdraw(2000)
s.bal()

#95

start=int(input())
stop=int(input())
even=[]
odd=[]
for i in range(start,stop):
    if i%2==0:
        even.append(i)
    else:
        odd.append(i)

print("even numbers:",even,end=" ")
print()
print("odd numbers:",odd,end=" ")
```

```
#96
string=str(input())
n=string.upper()
for i in range(len(n)-1,-1,-1):
    reverse=n[i]
    print(reverse,end=" ")
print()
```

```
#97
succesful_sales=0
loses=0
sales_report=[100,-1,-2,400,500,900,-9,-18,150,-3]
for i in sales_report:
    if i>0:
        succesful_sales+=1
    else:
        loses+=1
print("Number of succesful sales:",succesful_sales)
print("Number of loses:",loses)
```

```
#98)
arr=[]

average=[]

row=int(input("Enter the number of rows"))
coloumn=int(input("Enter the number of coloumn"))
for i in range(row):
    e=list(map(int,input().split()))
    arr.append(e)
print("average grade for each students")
print()
max_d=arr[0][0]
for i in range(row):#4
    total=0
    for j in range(coloumn):#3
        total+=arr[i][j]#85#
        avg=total/coloumn
    average.append(f"{avg:.2f}")
    print(f"{avg:.2f}",end=" ")
    print()
for i in range(coloumn):
    maximum=arr[0][i]
    for j in range(1,row):
        if arr[j][i]>maximum:
            maximum=arr[j][i]
    print(maximum)

total=0
for i in average:
    i=float(i)
```

```
    total+=i
    total_average=total/len(average)
print(f"The total average is {total_average:.2f}")

#99)
arr=[]
row=int(input("Enter the number of rows"))
coloumn=int(input("Enter the number of coloumn"))
for i in range(row):
    e=list(map(int,input().split()))
    arr.append(e)
total_l=[]
for i in range(row):
    total=0
    for j in range(coloumn):
        total+=arr[i][j]

    print(total)
    total_l.append(total)
check=int(input("enter to find the maximum quantity"))
maximum=max(arr[check-1])
check=arr[check-1].index(maximum)+1



print("the maximum quantity",chr(64+check))
minimum=min(total_l)
index=total_l.index(minimum)
print("the lowest quantity is",index+1)
```

```
#100)

d={"harini":[90,95,99,100,91],"rickshitha":[85,95,78,100,92],"hafeesha":[98,75,80,85,100]}
print("original dictionary:",d)
print()
d["harini"]=[90,95,99,100,94,91]
print("after updation:",d)
print()
d["poorani"]=[100,67,89,99,95]
print("after adding:",d)
print()
student_name=input()
if student_name in d:

    total=0

    for i in d[student_name]:
        total+=i
        avg=total/len(d[student_name])
    print("the average is :",avg)
    print()
else:
    print("students name not exits")
if student_name in d:
    d.pop(student_name)
    print("after deletion",d)
else:
    print("Students name does not exists")

#101)
t=(10,)
```

```
n=int(input("enter a number to multiply"))
print("after multiplication:",t*n)
for i in t:
    print(t.index(i))
l=list((t))
l[0]=25
l.extend([30,35,35,40,40])
t=tuple((l))
print("after the updation:",t)
new_string=" "
for i in t:
    new_string+=str((i))
print("the string value is",new_string)
maxi=0

for i in t:
    if i>maxi:
        maxi=i
print("the maximum element is:",maxi)
print("the minimum element is:",min(t))
e=int(input())
if e in t:
    print(t.count(e))
else:
    print("element does not exits")
new_t=((1,2,3,4),(5,6,7,8))
print("the nested tuple is ",new_t)
for i in new_t:
    print(i)
element_del=int(input())
if element_del in t:
```

```
    l=list((t))
    l.remove(element_del)
    t=tuple((l))
    print(t)
else:
    print("element does not exit")



#102)


r=int(input())
c=int(input())
arr=[]
for i in range(r):
    e=list(map(int,input().split()))
    arr.append(e)
for i in range(c):
    if i%2==0:
        for j in range(r):
            print(arr[j][i],end=" ")
    else:
        for j in range(r-1,-1,-1):
            print(arr[j][i],end=" ")




#103
r=int(input())
c=int(input())
arr=[]#123
     #456
```

```
    #789
for i in range(r):
    e=list(map(int,input().split()))
    arr.append(e)
for i in range(c):#3
    left=0#0
    right=i#0
    while left<r and right>=0:
        print(arr[left][right],end=" ")#00#01#10#02#11#20
        left+=1#1
        right-=1#-1
    print()
for i in range(1,r):
    left=i#1
    right=c-1#2
    while left<r and right>=0:
        print(arr[left][right],end=" ")#12#21#22
        left+=1
        right-=1
    print()



#104
n=int(input())#1234
rev=0
while n!=0:
    temp=n%10 #1234%10=4|123%10=3|12%10=2|1%10=1
    rev=(rev*10)+temp# 0*10+4=4|4*10+3=43|43*10+2=432|432*10+1=4321
    n=n//10 #1234//10=123|123//10=12|12//10=1|1//10=0
print(rev)
```

#105

```
n=int(input())
last_digit=n%10
mid_digit=(n//10)%10
first_digit=n//100
rev=last_digit*100+mid_digit*10+first_digit*1
print(rev)
```

#106

```
n=int(input())
n=str(n)
count=len(n)
d_sum=0
for i in n:
    d_sum=d_sum+int(i)**count

print(d_sum)
if d_sum==int(n):
    print("Amstrong Number")
else:
    print("Not a Amstrong number")
```

#107)

```
n=int(input())#1234
rev=0
while n!=0:
    temp=n%10 #1234%10=4|123%10=3|12%10=2|1%10=1
```

```
    rev=(rev*10)+temp# 0*10+4=4|4*10+3=43|43*10+2=432|432*10+1=4321
    n=n//10 #1234//10=123|123//10=12|12//10=1|1//10=0
print(rev)

#108
# another method for doing first question if three digit:
n=int(input())
last_digit=n%10
mid_digit=(n//10)%10
first_digit=n//100
rev=last_digit*100+mid_digit*10+first_digit*1
print(rev)



#109
n=int(input())
n=str(n)
count=len(n)
d_sum=0
for i in n:
    d_sum=d_sum+int(i)**count

print(d_sum)
if d_sum==int(n):
    print("Amstrong Number")
else:
    print("Not a Amstrong number")



#110)
```

```
n=int(input())#123
n=abs(n)
n=str(n)
sum_digits=0
for i in n:#1#2#3
    i=int(i)
    sum_digits=sum_digits+i#0+1=1#1+2=3#3+3=6

print(int(sum_digits))



#111)
n=int(input())
n=str(n)
product=1
for i in n:
    i=int(i)
    product=product*i
print(product)



#112)
n=int(input())
for i in range(n,0,-1):
    print(i,end=" ")

#113
d={}
def add():
    Name=input("Enter the new student name")
    Id=input("Enter the new student id")
```

```
    Grade=input("Enter the new student grade")
    Major=input("Enter the new student major")
    d["Name"]=Name
    d["Id"]=Id
    d["Grade"]=Grade
    d["Major"]=Major
    d[Id] = d
    print("added succesfuuly")
def update():
    Id=input("enter the student id")
    if Id in d:
        d[Id]["Name"]="Poorani"
        d[Id]["Grade"]="BCOM"
        d[Id]["Major"]="commerce"
        d[Id] = {"Name": "Poorani", "Grade": "BCOM", "Major": "commerce"}
        print("updated successfully")
    else:
        print("no student")

def display():
    for i in d.items():
        print(i)

    if len(d)==0:
        print("No student found")
def search():
    Id=input("enter the student id for search")
    if Id in d:
        print(d[Id])
    else:
        print("not found")
```

```
def remove():
    Id=input("Enter the student id to be removed")
    if Id in d :
        del d[Id]


        print("student removed successfully")
    else:
        print("no student found in id:")

def bye():
    print("goodbye")
    quit()

while True:
    print("1)add 2)update 3)search 4) remove 5)exit")
    option=int(input("enter the option"))
    match option:
        case 1:
            add()
        case 2:
            update()
        case 3:
            search()
        case 4:
            remove()
        case 5:
            bye()
        case _:
            print("invalid option")
```

#114

TICKET BOOKING CHATBOT

```
d={"train1" :{"train name": "cholan express", "tickets available": 50 ,"waiting list":"as far as no waiting list"},
  "train2":{"train name":" kovai express","tickets available":0,"waiting list":20},
  "train3":{"train name": "kachiguda express","tickets avilable":20, "waiting list":"as far as no waiting list"}}
def check_availability(train_number):
   result=d.get(train_number)
   if result:
      return f" your train name is {result['train name']} with tickets availability {result['tickets available']}"
   else:
      return "train not available"
def check_waiting_list(train_number):
   result=d.get(train_number)
   if result:
      return f" your waiting list is {result['waiting list']}"
def chatbot():
   print("Welcome to the train check chatbot")
   train_number=input()
   print("how can i assit you?")
   print("1)train name with ticket,2)train waiting list")
   choice=int(input("enter the choice 1/2"))
   if choice==1:
      print(check_availability(train_number))
   if choice==2:
      print(check_waiting_list(train_number))
chatbot()
```

#115

quiz contest

```
print("Welcome to the cricket quiz")

print("1)questions 2)answers 3) winner")

score=0

def questions():

    questions=["1)When did india won the t20 world cup recently, 2) who is the current captain of indian cricket team ,3) who is known as god of cricket,4) who is called the captain  cool ,5) who is the king"]

    print(questions)

answers=["2024","rohit sharma","sachin tendulkar" , "mahendra singh dhoni","virat kholi"]

def answer():

    global score

    answer1=input("enter the answer for question1 ")

    answer2=input("enter the answer for question2")

    answer3=input("enter the answer for question3")

    answer4=input("enter the answer for question4")

    answer5=input("enter the answer for question5")

    if answers[0]==answer1:

        score+=1

    if answers[1]==answer2:

        score+=1

    if answers[2]==answer3:

        score+=1

    if answers[3]==answer4:

        score+=1

    if answers[4]==answer5:

        score+=1

def winner():

    global score

    if score==5:

        print("you won the quiz")

    else:

        print(f"sorry you have loose by {(5-score)}")
```

```
while True:
    choice=int(input("enter number from 1-3 orderly"))
    match choice:
        case 1:
            questions()
        case 2:
            answer()
        case 3:
            winner()
        case _:
            print("game over")

#116
n=int(input())
sum_digits=0
num=n
while n>0:
    last=n%10
    sum_digits+=last
    n=n//10
if num%sum_digits==0:
    print(f"{num} is a Harshad Number")
else:
    print(f"{num} is not a Harshad Number")



#117
n=int(input("Enter number of items in the list"))
count=0
book_name=[]
while n>0:
```

```
    s=int(input())
    book_name.append(s)
    n=n-1
for i in book_name:#[1,2,3]
    i=int(i)
    count=i+count#0+1=1#1+2=3#3+3=6
print("The total number of books read by all students is:",count)

#118

rows=int(input())
for i in range(1,rows+1,1):
    for j in range(i):
        print("*",end="")
    print()

#119

import math
n=int(input())
if n<0:
    print("Invalid number for finding squares:")
else:
    root=math.sqrt(n)
    if root**2==n:
        print(f"{n} is a Perfect square")
    else:
        print(f"{n} is not a perfect square")



#120
```

```
rows=int(input())
coloums=int(input())
for i in range(0,rows):
    for j in range (0,coloums):
        if i==0 or i==rows-1 or j==0 or j==coloums-1:
            print("1",end=" ")
        else:
            print("0",end=" ")
    print()

#121

a=1
n=int(input())
for i in range(1,n,1):
    i=str(i)
    print("*".join(a*i))
    a=a+1
b=a-1
for j in range(n-1,0,-1):
    j=str(j)
    print("*".join(b*j))
    b=b-1

#122) question

n=int(input())
a=int(input())
n1=0
n2=1
count=0
```

```
l=[]
if n<=0:
   print("Enter a valid number")
if n==1:
   l.append(n1)
while count<n:
   l.append(n1)
   n3=n1+n2
   n1=n2
   n2=n3
   count+=1
even=[]
for i in l:
   i=int(i)
   if i%2==0:
      even.append(i)
for i in even:
   b=even[a]
print(b)
```

```
#123)
import math
n=int(input())
count=0
plots=[]# 25 36
for i in range(n):
   area=int(input())
   plots.append(area)
for i in plots:
```

```
    i=int(i) #25
    sqrt=int(math.sqrt(i)) #5
    if i==(sqrt**2):
        count+=1 #1
print(count)



#124)without list
x=int(input())
count=0
n1=1
n2=1
while count<x:
    n3=n1+n2
    n1=n2
    n2=n3
    if n2%2==0:
        count+=1
print(n2)

#125
try:
    main_string=str(input())
    search_word=str(input())
    index=main_string.find(search_word)
    if index>=0:
        print(index)
    if index==-1:
        raise ValueError
except ValueError :
```

```
    print("Substring not found")



#126


try:
  n=int(input("ENTER NUMBER OF ELEMENTS IN THE LIST"))
  l=[]
  avg=0
  sum_of_elements=0
  for i in range(n):
     l.append(int(input()))
  for i in l:
     sum_of_elements+=i
     if len(l)>0:
        avg=sum_of_elements/n
  print(avg)
  if len(l)==0:
    raise ZeroDivisionError
except ZeroDivisionError:
    print("List cant be empty")


 #127
try:
    list_items=["name","class","date of birth","admission number"]
    search=int(input())
    length=len(list_items)
    if search<=length-1:
       print(list_items[search])
    if search>=length:
        raise IndexError
```

```
except IndexError as e:
    print("Index Error")



#128)


try:
    age=int(input())
    assert age>=18 ,"Participants should be atleast 18 years"
    assert age<60,"Participants should be not older than 60"
    print("Valid for the event")
except AssertionError as e:
    print("Registration error:",str(e))

#2)second question

try:

   n=int(input())
   n2=int(input())
   sum_num=n+n2
   print(sum_num)
except ValueError as e:
    print(e)

#129

class NegativeFundError(Exception):
    pass
class InsufficientFundError(Exception):
```

```
    pass
try:
        amount=int(input("Enter withdraw amount:"))
        balance=10000
        if amount>balance:
            raise InsufficientFundError("InsufficentFundError,your amount is greater than balance")
        if amount<0:
            raise NegativeFundError("Negative fund error ,the amount is negative")
        else:
            remain=balance-amount
            print("You have succesfully withdrawn and the remaining amount is ",remain)
except InsufficientFundError as e:
            print("Error:",str(e))
except NegativeFundError as e:
        print("Error", str(e))
```

#130)QUESTION

```
import array as arr
sales_figure=arr.array('i',[1200,1300,1400,1500])
print(sales_figure[3])
list_sales_figure=sales_figure.tolist()
list_sales_figure.reverse()
rev_sales=arr.array('i',list_sales_figure)
for i in rev_sales:
    print(i,end=" ")
print()
list_sales_figure=sales_figure.tolist()
list_sales_figure.insert(2,1600)
```

```
list_sales_figure.insert(3,1700)
new_sales=arr.array('i',list_sales_figure)
for i in new_sales:
   print(i,end=" ")



#131)QUESTION
import array as arr
marks=arr.array('i',[100,70,80,35,40,85,45,60,74,33])
print("marks of student is:")
for i in marks:
    print(i,end=" ")
print()
list_marks=marks.tolist()
list_marks.remove(35)
list_marks.insert(3,50)
array_marks=arr.array('i',list_marks)
print("Marks after updation:")
for i in array_marks:
   print(i,end=" ")
print()
for i in array_marks:
   highest=100
   if i==highest:
      print(f" Your mark is {highest} you are having a great mark,keep it up")
   lowest=33
   if i==lowest:
      print(f"Your mark is {lowest}you should improve by continuous learnig")



#132)QUESTION
```

```
import array as arr
prices=arr.array('d',[10.99,5.49,20.00,7.95,12.75])
sum_prices=0
for i in prices:
    sum_prices+=i
print(f"The total cost is :{sum_prices:.2f}")
list_prices=prices.tolist()
list_prices.remove(5.49)
array_prices=arr.array('d',list_prices)
print("prices after removal of 2nd item")
for i in array_prices:
    print(i,end=" ")
print()
list_prices=array_prices.tolist()
list_prices.insert(1,6.00)
array_prices=arr.array('d',list_prices)
print("prices after updation")
for i in array_prices:
    print(i,end=" ")
print()

#133
import calendar
year=2024
month=10
print(calendar.month(year,month))
'''print(calendar.calendar(year))'''
is_leap=calendar.isleap(year)
print(f"{year} is a leap year:{is_leap}")
#134
import calendar
```

```
year=2024
month=10
first_weekday,num_days=calendar.monthrange(year,month)
print(f"First weekday:{first_weekday},Number of days:{num_days}")

#135
import calendar
year=2024
month=10
for day in calendar.Calendar().itermonthdays(year,month):
    print(day)
#136
import calendar
text_cal=calendar.TextCalendar(calendar.SUNDAY)
year=2024
month=10
plain_text_cal=text_cal.formatmonth(year,month)
print(plain_text_cal)

#137
from datetime import datetime,timedelta
now=datetime.now()
print(f"Current date and time :{now}")

days_in_past=5
hours_in_past=3
minutes_in_past=30
past_time=now-timedelta(days=days_in_past,hours=hours_in_past,minutes=minutes_in_past)
print(f"Past date and time (5 days,3 hours, 30 minutes ago):{past_time}")

#138
```

```
import string
sentence="hello world from python"
capitalized_sentence=string.capwords(sentence)
print(capitalized_sentence)



#CW QUESTION

#1
def withdraw(w):
    global balance
    balance-=w
    print("withdrawn successfully.Go back to the main menu to view balance")
def deposit(d):
    global balance
    balance+=d
    print("deposited successfully")
def checkbalance():
    print("Your balance is",balance)
def end_transaction():
    quit()

balance=20000
print("********RRR Bank********")
while True:
    print("Withdraw \n2.Deposit\n3.balance\n4.Exit")
    option=int(input("enter the transaction to proceed with"))
    match option:
        case 1:
            draw=int(input())
```

```
            withdraw(draw)
        case 2:
            dep=int(input())
            deposit(dep)
        case 3:
            checkbalance()
        case 4:
            end_transaction()
        case _:
            print("enter a valid option")

#2
import random
import secrets
random.seed(2)
print(random.random())
l=[1,2,3,4,5]
print(random.choices(l,k=2))
random.shuffle(l)
print(l)
print(random.randrange(1,20))
print(random.choice(l))
print(secrets.choice(l))
print(random.uniform(1,10))



#3
#non parameterised constructor

class Student:
    def __init__(self):
```

```
        self.name="Mithran"
        self.dept="Ai"
    def display(self):
        print(f"Name={self.name}\nDepartment={self.dept}")
s=Student()#object creation
s.display()



class Library:
    def __init__(self):
        self.bookname="Ponniyin selvan"
        self.authorname="Kalki"
    def displaybookdetails(self):
        print(f"Book name :{self.bookname}\nAuthor name:{self.authorname}")
s=Library()
s.displaybookdetails()




#parameterised constructor


class Student:
    def __init__(self,name,dept):
        self.name=name
        self.dept=dept
    def display(self):
        print(f"Name={self.name}\nDepartment={self.dept}")
s=Student("Harini","Artificial intelligence")
s.display()
```

```
#default value
class Student:
    def __init__(self,dept,name="Harini"):
        self.name=name
        self.dept=dept
    def display(self):
        print(f"Name:{self.name}\nDepartment:{self.dept}")
s=Student("artificial intelligence")
s.display()



#destructor


class Student:
    def __init__(self,dept,name="Harini"):
        self.name=name
        self.dept=dept
    def display(self):
        print(f"Name:{self.name}\nDepartment:{self.dept}")
    def __del__(self):
        print("object deleted")# can use pass also so after display a blank statement appear instead of object deleted
s=Student("AI")
s.display()
del s
#s.display() of we again call name error will be shown as there is no object and it is deleted



#4
#functions with argument without return type
def sum(a,b):
```

```
    result=a+b
    print(result)
a=int(input())
b=int(input())
sum(a,b)

#functions with argument with return type
def sum(a,b):
    result=a+b
    return result
a=int(input())
b=int(input())
ans=sum(a,b)
print(ans)

#functions without argument without return type
def sum():
    a=int(input())
    b=int(input())
    result=a+b
    print(result)
sum()

#functions without argument with return type:
def sum():
    a=int(input())
    b=int(input())
    result=a+b
    return result
ans=sum()
print(ans)
```

```
#5
# Single inheritance

class Student:# parent class
    def display(self):
        print("base class parent")
class Student_derived(Student): #child class
    def show(self):
        print("derived class child")
s=Student_derived()# for inheritance we have to create object for the derived class
s.display()
s.show()



# with variables
class Person:
    def __init__(self,name,age):
        self.name=name
        self.age=age
    def display(self):
        print(f"Name={self.name}\nAge={self.age}")

class Student(Person):
    def __init__(self,name,age,stu_id,stu_dept):# instance varialble
        super().__init__(name,age)
        self.stu_dept=stu_dept
        self.stu_id=stu_id
    def printDetails(self):
        self.display()
        print(f"ID={self.stu_id}\nDepartment{self.stu_dept}")
```

```
s=Student('Geetha',35,1200,'AI')
s.printDetails()



#6
import datetime
current_datetime=datetime.datetime.now()
current_date=datetime.date.today()
print(f"current date time={current_datetime}")
current_datetime=datetime.datetime.today()
print("current date time",current_datetime)
specific_date=datetime.date(2023,9,23)
print("specific date",specific_date)
specific_time=datetime.time(14,30,45)
print("specific date",specific_time)
specific_datetime=datetime.datetime(2024,9,23,14,30,45)
print("specific date time:",specific_datetime)
formatted_date=current_datetime.strftime("%Y-%m-%d %H:%M:%S")
print("formatted date and time:",formatted_date)
date_string="2024-09-23 14:30:45"
parsed_date=datetime.datetime.strptime(date_string,"%Y-%m-%d %H:%M:%S")
print("parsed datetime object:",parsed_date)
ten_days=datetime.timedelta(days=10)
date_10_days_ago=current_date-ten_days
print("date 10 days ago:",date_10_days_ago)
date_10_days_later=current_date+ten_days
print("date 10 days later:",date_10_days_later)
date1=datetime.date(2024,10,8)
date2=datetime.date(2006,10,5)
difference=date1-date2
```

```
print("difference beteween two dates:",difference.days,"days")


#7
import pytz
current_utc=datetime.datetime.now(pytz.utc)
print("current utc time:",current_utc)
eastern=pytz.timezone('US/Eastern')
eastern_time=current_utc.astimezone(eastern)
print("eastern time:",eastern_time)




#defining a simple class
class Student:
    s_name="harini"
    s_dept="ai"
s=Student() #object creation->objectname=classname()
print(f"Name={s.s_name} \n Department={s.s_dept}")



#defining outside
class Student:
    s_name=""
    s_dept=""
s=Student()
s.s_name="Geetha"
s.s_dept="ai"
print(f"Name={s.s_name} \n Department={s.s_dept}")



#creating two objects
```

```
class Student:
    s_name=""
    s_dept=""
s1=Student()
s2=Student()
s1.s_name="Geetha"
s1.s_dept="ai"
s2.s_name="mithran"
s2.s_dept="ai"
print(f"Name={s1.s_name} \n Department={s1.s_dept}")
print("assign based on object 2")
print(f"Name={s2.s_name} \n Department={s2.s_dept}")
```

```
#inside a method
class Student:
    name="harini"
    dept="ai"
    def display(self):#the first keyword is self
        print(f"Name={self.name} \n Department={self.dept}")
s=Student()
s.display()# method or function call
```

```
#using constructor
class Student:
    def __init__(self,name=" "):#should contain space after def for defining the constructor
        self.name=name
    def display(self):
```

```
        print("Name=",self.name)
s=Student("Geetha")
s.display()

# using user input:
class Student:
    def __init__(self,name=" "):
        self.name=name
    def display(self):
        print("Name=",self.name)
s_name=input()
s=Student(s_name)
s.display()

# user input inside the class
class Student:
    def __init__(self,name=" "):
        self.name=name
    def getuserinput(self):
        self.id=input("enter the id")
    def display(self):
        print("Name=",self.name)
        print("id=",self.id)
s_name=input("enter the name")
s=Student(s_name)
s.getuserinput()
s.display()

#8

import random
```

```
user_count=0
system_count=0
l=[1,2,3]
while True:
    print("1.rock,2.paper,3.scissior")
    user=int(input("enter any option:"))
    system=random.choice(l)
    print(system)

    if user ==1 and system==1 or user==2 and system==2 or user==3 and system==3:
        print("tie")
    if user==1 and system==3 or user==2 and system==1 or user==3 and system==2:
        print("user wins")
        user_count+=1
    if user==1 and system==2 or user==2 and system==3 or user==3 and system==1:
        print("system wins")
        system_count+=1
if user_count>system_count:
    print("user wins finally")
else:
    print("system wins finally")
```